

**SISTEM REKOMENDASI PEMILIHAN DOSEN PEMBIMBING  
TUGAS AKHIR DENGAN METRIK *COSINE SIMILARITY***



**Disusun sebagai salah satu syarat memperoleh Gelar Strata I  
pada Jurusan informatika Fakultas Komunikasi dan Informatika**

**Oleh:  
ZULFA FAJRUL FALAH  
L200170149**

**PROGRAM STUDI INFORMATIKA  
FAKULTAS KOMUNIKASI DAN INFORMATIKA  
UNIVERSITAS MUHAMMADIYAH SURAKARTA  
2021**

**HALAMAN PERSETUJUAN**

**SISTEM REKOMENDASI PEMILIHAN DOSEN PEMBIMBING  
TUGAS AKHIR DENGAN METRIK *COSINE SIMILARITY***

**PUBLIKASI ILMIAH**

oleh:

**ZULFA FAJRUL FALAH**  
**L200070149**

Telah diperiksa dan disetujui untuk diuji oleh:  
Dosen Pembimbing

  
**Fajar Suryawan, S.T., M.Eng.Sc., Ph.D.**

**NIK. 942**

*seni proposal.*

*22/4/2021*

## HALAMAN PENGESAHAN

# SISTEM REKOMENDASI PEMILIHAN DOSEN PEMBIMBING TUGAS AKHIR DENGAN METRIK *COSINE SIMILARITY*

OLEH

ZULFA FAJRUL FALAH

L200170149

Telah dipertahankan di depan Dewan Penguji  
Fakultas Komunikasi dan Informatika  
Universitas Muhammadiyah Surakarta  
Pada hari Jum'at, 12 November 2021

dan dinyatakan telah memenuhi syarat

Dewan Penguji:

1. Fajar Suryawan, ST, M.EngSc., Ph.D.

(Ketua Dewan Penguji)



2. Maryam, S.Kom., M.Eng.

(Anggota I Dewan Penguji)




3. Dimas Aryo Anggoro, S.Kom., M.Sc..

(Anggota II Dewan Penguji)



Dekan  
Fakultas Komunikasi dan Informatika



  
Nurchayati, S.T., M.Sc., Ph.D.  
NIK.881

## **PERNYATAAN**

Dengan ini saya menyatakan bahwa dalam publikasi ilmiah ini tidak terdapat karya yang pernah diajukan untuk memperoleh gelar kesarjanaan di suatu perguruan tinggi dan sepanjang pengetahuan saya juga tidak terdapat karya atau pendapat yang pernah ditulis atau diterbitkan orang lain, kecuali secara tertulis diacu dalam naskah dan disebutkan dalam daftar pustaka.

Apabila kelak terbukti ada ketidakbenaran dalam pernyataan saya di atas, maka akan saya pertanggungjawabkan sepenuhnya.

**Surakarta, 12 November 2021**

Penulis



**ZULFA FAJRUL FALAH**

**L200170149**

# **SISTEM REKOMENDASI PEMILIHAN DOSEN PEMBIMBING TUGAS AKHIR DENGAN METRIK *COSINE SIMILARITY***

## **Abstrak**

Pemilihan dosen pembimbing merupakan hal yang penting dan merupakan salah satu faktor penentu lancar tidaknya suatu penelitian tugas akhir mahasiswa. Pada saat ini pemilihan dosen pembimbing di lokasi penelitian ini dilakukan secara manual. Mahasiswa sendiri yang melakukan pemilihan dosen pembimbing berdasarkan *research* yang bersumber dari data yang ada. Ada yang bersumber dari teman satu angkatan, ada yang bersumber dari kakak tingkat yang sudah lulus, bahkan ada yang memilih dosen pembimbing tanpa suatu alasan apakah dosen yang dipilih tersebut cocok atau tidak dengan minat dan dengan topik tugas akhir yang akan diambil. Pemilihan dosen pembimbing yang tepat akan memberikan dampak yang besar terhadap mahasiswa, mahasiswa akan lebih semangat dalam mengerjakan tugas akhir dan dapat dimudahkan dalam melakukan penelitian karena minat mahasiswa dan konsentrasi dosen sudah cocok. Dalam penelitian ini minat mahasiswa dengan minat dosen ditentukan kecocokannya berdasarkan pengolahan data yang diinputkan oleh mahasiswa berupa judul, abstrak dan keyword. Data ini dicocokkan dengan judul, abstrak dan keyword artikel-artikel calon dosen pembimbing yang telah terbit di Google Scholar. Sistem rekomendasi ini menggunakan metode *content-based filtering* untuk menghasilkan daftar rekomendasi pembimbing tugas akhir. Kemudian menggunakan algoritma cosine similarity untuk menghitung seberapa mirip topik yang diajukan mahasiswa dengan penelitian dosen. Dalam membangun sistem rekomendasi yang berbasis website, penulis menggunakan dua *web framework* Django sebagai backend dan ReactJs sebagai frontend. Sistem rekomendasi ini diharapkan menghasilkan rekomendasi dosen pembimbing tugas akhir yang memiliki minat serta keahlian yang cocok dengan topik tugas akhir yang akan diajukan oleh mahasiswa.

**Kata Kunci:** *Cosine Similarity*, sistem rekomendasi, *web scraping*, *content-based filtering*.

## **Abstract**

The selection of a supervisor is an important thing and is one of the determinants of whether or not a student's final project research is successful. At this time the selection of supervisors at the location of this research is done manually. Students themselves make the selection of supervisors based on research sourced from existing data. Some come from classmates, some come from seniors who have graduated, some even choose a supervisor without any reason whether the chosen lecturer is suitable or not with interests and with the topic of the final project to be taken. The selection of the right supervisor will have a big impact on students, students will be more enthusiastic in doing the final project and it can be facilitated in conducting research because student interest and lecturer concentration are matched. In this study, the student's interest with the lecturer's interest was determined based on the data processing entered by the student in the form of title, abstract and keywords. This data is matched with the titles,

abstracts and keywords of articles for prospective supervisors that have been published on Google Scholar. This recommendation system uses the *content-based filtering* method to produce a list of recommendations for the final project supervisor. Then use the *cosine similarity* algorithm to calculate how similar the topic proposed by students is to the lecturer's research. In building a website-based recommendation system, the author uses two web frameworks django as the backend and ReactJs as the frontend. This recommendation system is expected to produce recommendations for final project supervisors who have interests and expertise that match the topic of the final project that will be submitted by students.

**Keywords:** *Cosine Similarity, recommendation system, web scraping, content-based filtering.*

## 1. PENDAHULUAN

Tugas akhir merupakan syarat lulusnya mahasiswa dari sebuah universitas, tugas akhir juga menjadi karya ilmiah terakhir seorang mahasiswa untuk menutup masa studinya dalam menempuh pendidikan strata satu. Dalam perancangan tugas akhir mahasiswa didampingi oleh dosen pembimbing, dosen pembimbinglah yang akan menjadi partner dalam kolaborasi antara mahasiswa dengan dosen dalam mengerjakan penelitian yang sudah diajukan. Fungsi dosen pembimbing juga sebagai fasilitator untuk mahasiswa jika mahasiswa mengalami kesulitan atau keraguan dalam proses penelitian. Dosen pembimbing juga harus menguasai bidang yang sesuai dengan topik yang diambil oleh mahasiswa agar hasil penelitian menjadi maksimal. Oleh sebab itu peran seorang dosen pembimbing sangatlah penting.

Di prodi tempat penelitian ini berlangsung, pemilihan dosen pembimbing dilakukan secara manual dan mandiri oleh mahasiswa. Mahasiswa memilih langsung dosen pembimbingnya ketika masa pengisian kartu rencana studi berlangsung. Pemilihan dosen pembimbing umumnya didasari oleh pengetahuan personal terkait dengan spesialisasi dosen dan juga didasari oleh riset yang dilakukan sendiri oleh mahasiswa dengan sumber data seadanya, ada yang bersumber dari teman satu angkatan atau dari kakak tingkat yang sudah lulus. Bahkan ada mahasiswa yang memilih dosen pembimbing tanpa suatu alasan khusus; apakah dosen tersebut cocok dengan minat atau dengan topik yang akan diajukannya.

Dari permasalahan di atas, penulis melihat urgensi dibangunnya suatu sistem rekomendasi untuk membantu mahasiswa menentukan dosen pembimbing. Sebuah sistem rekomendasi berfungsi untuk memilah-milah sejumlah besar data untuk

mengidentifikasi minat pengguna dan mempermudah dalam menemukan informasi dan mempermudah dalam menentukan keputusan (Nagarnaik & Thomas, 2015). Suatu sistem rekomendasi juga merupakan sistem penyaringan informasi yang digunakan untuk memprediksi peringkat atau preferensi yang akan diberikan kepada pengguna pada suatu item seperti musik, buku, film dan dokumen. Model sistem rekomendasi dapat dibangun dari karakteristik suatu item (*content-based filtering*) atau dengan pendekatan lingkungan pengguna (*collaborative filtering approaches*) (Sharma & Gera, 2013). Ini juga yang akan dipakai dalam penelitian ini.

*Content-based filtering* bekerja dengan pemfilteran berbasis konten yang menggunakan fitur item untuk merekomendasikan item lain yang *similar* dengan apa yang disukai pengguna. Ini juga merupakan salah satu teknik rekomendasi yang paling sukses, yang didasarkan pada korelasi antar konten, dia menggunakan informasi item yang direpresentasikan sebagai atribut untuk menghitung kesamaan antar item (Son & Kim, 2017). Namun strategi ini juga mempunyai kelemahan, yakni metode ini tidak dapat menghasilkan rekomendasi yang sesuai jika konten yang dianalisis untuk suatu item tidak mengandung informasi yang sesuai untuk kategorisasi (Debnath et al., 2008). Ada beberapa metode yang dapat digunakan untuk menghitung similaritas antar konten seperti euclidean distance, *cosine similarity* dan manhattan distance. Penelitian yang dilakukan oleh Fathin pada tahun 2019 yang membandingkan hasil perhitungan antara *cosine similarity* dan *euclidean distance* menunjukkan nilai similaritas dan memiliki tingkat akurasi yang sama (Mubarak, 2019). Dalam penelitian ini metode yang digunakan dalam membandingkan similaritas antar konten menggunakan *cosine similarity*.

Sudah ada beberapa penelitian yang dilakukan dengan topik pemilihan dosen pembimbing tugas akhir. Seperti yang dilakukan oleh Asrul pada tahun 2018, dalam penelitian nya Asrul menggunakan metode *Analytical Hierarchy Process* (AHP) untuk membangun sistem pendukung keputusan untuk memilih dosen pembimbing. Pada metode AHP pemberian bobot pada kriteria dilakukan oleh para ahli, yang mana hal tersebut sangat bersifat subjektif karena pemberian nilai kriteria tergantung pada setiap ahli (Abdullah & Pangestika, 2018). Kemudian terdapat juga penelitian yang dilakukan di Departemen Ilmu Komputer/Informatika Fakultas Sains dan Matematika Universitas Diponegoro mengenai sistem pemilihan dosen pembimbing, dalam penelitian tersebut menggunakan *Vector Space Model* (VSM) sebagai metode pencocokan string judul

penelitian mahasiswa dan penelitian yang pernah dilakukan oleh dosen (Amalina & Sutikno, 2017).

Penelitian ini menggunakan data publikasi dosen yang telah terbit di halaman Google Scholar. Pengambilan data dilakukan dengan melakukan *web scraping* pada halaman profil google scholar setiap kandidat dosen pembimbing. Data ini akan menjadi basis dalam membangun model sistem rekomendasi dan yang kemudian akan menghasilkan informasi yang berguna dalam menentukan dosen pembimbing yang cocok untuk mahasiswa. Data yang ambil dari google scholar antara lain judul, abstrak dan keyword dari publikasi dosen. Sepanjang sepengetahuan penulis, penelitian ini merupakan penelitian pertama di indonesia yang yang memanfaatkan data di google scholar untuk membangun sistem rekomendasi. Tujuan dari sistem rekomendasi ini adalah menghasilkan daftar dosen pembimbing tugas akhir yang sesuai dengan topik yang diajukan oleh mahasiswa, dan dapat mempermudah mahasiswa dalam memilih dosen pembimbing yang cocok berdasarkan topik yang diajukan.

## **2. METODE**

Dalam melakukan penelitian ini, pertama-tama kita mengumpulkan data publikasi dari Google Scholar dilanjutkan dengan penyiapan awal data. Data yang sudah siap lalu diumpankan ke algoritma utama. Sebuah website dibangun untuk menanam sistem rekomendasi tersebut, yang menjadi interface utama antara sistem dengan pengguna. Detailnya adalah sebagai berikut.

### **2.1 Data Collection**

Data diambil dengan melakukan *web scraping* pada halaman profil google scholar tiap dosen pembimbing. *Web scraping* merupakan proses pengambilan sebuah dokumen semi - terstruktur dari internet, yang pada umumnya berupa halaman - halaman web dalam bahasa markup seperti HTML atau XHTML, dan menganalisa dokumen tersebut untuk diambil data tertentu dari halaman tersebut untuk digunakan bagi kepentingan lain. Pada penelitian ini penulis menggunakan Python untuk melakukan *web scraping* dengan dibantu third party library seperti *Selenium*, *Beautifulsoup*, *Requests*, dan *CSV*, yang kemudian data-data hasil *web scraping* tadi disimpan kedalam file *CSV* untuk diproses ke tahap selanjutnya. Data set ini memiliki 603 baris dan 4 atribut. Lihat Tabel 1.



Tabel 1. Keterangan atribut data penelitian dosen pembimbing

No	Atribut	Keterangan
1	Nama	Nama dosen pembimbing
2	Judul	Judul penelitian
3	Abstrak	Abstrak penelitian
4	Keyword	Keyword penelitian

## 2.2 Data Preprocessing

Data Preprocessing merupakan salah satu tahap utama dalam proses penemuan *knowledge*, meskipun tahap ini tidak sepopuler tahap-tahap lain seperti *data mining*, data preprocessing sebenarnya melibatkan lebih banyak waktu dan usaha dalam seluruh proses analisis datanya (Pyle, 1999). Data hasil *web scraping* pada umumnya berupa data yang masih kotor atau data mentah, seperti masih terdapat elemen HTML yang tidak sengaja diambil, terdapat karakter non alfanumerik, dan baris data kosong atau tidak lengkap. Adanya inkonsistensi data dan noise yang terdapat pada dataset dapat mempengaruhi kinerja dari mesin (Jianqiang et al., 2018). *Data preprocessing* juga berfungsi untuk memperbaiki format data serta membersihkan dari gangguan atau *noise* pada data mentah (Gunawan, 2016). Berikut ini adalah beberapa langkah data preprocessing yang dilakukan di sini.

### 2.2.1 Punctuation Removal

Proses ini menghapus karakter yang bukan termasuk kedalam huruf-huruf alfabet karena karakter lain seperti tanda baca dan karakter non alfanumerik (kecuali spasi) seperti `!"#$%&'()*+,-./:;<=>?@[\\]^_`{|}~` dapat mempengaruhi hasil akurasi analisis. Pada penelitian ini penulis menggunakan python untuk membersihkan tanda baca dan karakter non alfanumerik.

### 2.2.2 Case Folding

Proses untuk mengubah semua huruf besar (uppercase) pada dokumen menjadi huruf kecil (lowercase).

### 2.2.3 Tokenization

*Tokenization*. Ini merupakan proses untuk membagi atau memecah teks yang berupa kalimat, paragraf ataupun dokumen menjadi token. Dalam ilmu linguistik token

merupakan unit terkecil dalam suatu teks. Token ini akan membantu untuk memahami konteks dan untuk pengembangan model NLP. *Tokenization* juga akan berguna untuk menafsirkan makna teks dengan menganalisis urutan kata-kata.

Tabel 2. Proses Tokenisasi

Teks Asli	Setelah Tokenisasi
'rancang bangun aplikasi pembelajaran hadist'	'rancang', 'bangun', 'aplikasi', 'pembelajaran', 'hadist'

#### 2.2.4 Stopwords Removal

Pada umumnya di dalam sebuah teks terdapat kata-kata yang paling umum muncul seperti preposisi, konjungsi, kata ganti, dan lain-lain. Kata-kata ini tidak banyak memberikan informasi ke dalam teks. Penghapusan kata yang kurang informatif dari teks dapat memberikan lebih banyak fokus kepada mesin untuk memproses kata-kata yang penting saja, dengan kata lain penghapusan kata yang kurang informatif tidak akan memberikan dampak negatif pada vektor, dan akan lebih efisien dari sisi pemrosesan.

#### 2.2.5 Word Vectorization

Merupakan Metodologi dalam NLP untuk memetakan kata atau frasa dari kosakata kedalam sebuah vektor bilangan real dan untuk menentukan prediksi kata atau kesamaan kata/semantik. Dengan melakukan *word vectorization* pada teks penelitian dosen pembimbing, mesin dapat memprosesnya sebagai vektor bilangan real bukan lagi sebagai sebuah kumpulan kata-kata. Untuk merepresentasikan sebuah dokumen perlu mengubahnya menjadi bentuk vektor, agar dapat diproses oleh mesin (Roul et al., 2017). Penggunaan skema *Term Frequency* (TF) dan *Inverse Document Frequency* (IDF) telah terbukti sebagai algoritma yang kuat dalam memproses data teks atau keperluan yang lainnya (Robertson, 2004). TF-IDF menggunakan frekuensi kata dan frekuensi dokumen untuk menghasilkan kata yang berbobot yang digunakan untuk merepresentasikan dokumen (Roul et al., 2017). Syarat frekuensi kata atau frekuensi dokumen pada pendekatan TF-IDF biasanya digunakan untuk menimbang masing-masing kata dalam dokumen teks yang sesuai dengan keunikannya (Zhang Yun-tao, Gong Ling, 2005). Kemudian nilai TF-IDF dapat dihitung dengan menggunakan persamaan:

## 2.3 Data Processing

### 2.3.1 Recommendation System

Sistem rekomendasi memiliki beberapa algoritma seperti *content-based filtering*, *collaborative filtering* dan kombinasi antara keduanya (Nilashi et al., 2013) (Nagarnaik & Thomas, 2015). Pada penelitian ini penulis menggunakan algoritma *content-based filtering* sebagai metode untuk menentukan hasil rekomendasi dari dosen pembimbing. Basis konten yang dipakai adalah teks dari judul, abstrak, dan keyword penelitian dosen. Sistem rekomendasi dengan teknik ini memiliki kesamaan dengan teknik lainnya dalam hal deskripsi item, profil pengguna, dan teknik untuk membandingkan profile dengan item untuk mengidentifikasi hasil rekomendasi yang paling sesuai untuk pengguna. (Sharma & Gera, 2013). Salah satu metode yang dipakai untuk mengukur kedekatan antar teks adalah metode *cosine similarity*, yang akan dipakai di sini.

### 2.3.2 Cosine Similarity

*Cosine Similarity* adalah metode yang digunakan untuk mengukur kemiripan antara dua dokumen teks yang dianggap sebagai vektor (Samuel et al., 2018). *Cosine similarity* juga merupakan metrik yang diimplementasikan secara luas dalam pencarian informasi dan sering diterapkan dalam membandingkan kemiripan dua teks (kalimat, paragraf atau seluruh dokumen), kemiripan antara dua dokumen didapat dengan menghitung nilai cosinus antara vektor-vektor antara dokumen (Rahutomo et al., 2012). Pada penelitian ini metode untuk menghitung kemiripan antara penelitian dosen dengan topik yang akan diajukan mahasiswa adalah dengan membandingkan similaritas judul, abstrak, keyword dari penelitian dosen pembimbing dengan judul, abstrak, keyword yang akan diajukan oleh mahasiswa. Nilai cosine similarity antar vektor dapat dihitung dengan persamaan berikut:

$$\text{Cos } \alpha = \frac{\mathbf{A} \cdot \mathbf{B}}{|\mathbf{A}| |\mathbf{B}|} = \frac{\sum_{i=1}^n A_i \times B_i}{\sqrt{\sum_{i=1}^n (A_i)^2} \times \sqrt{\sum_{i=1}^n (B_i)^2}} \quad (1)$$

Keterangan:

A = Vektor A, merupakan vektor penelitian dosen

B = Vektor B, merupakan vektor topik yang diajukan oleh mahasiswa

A • B = dot product antara vektor A dan vektor B

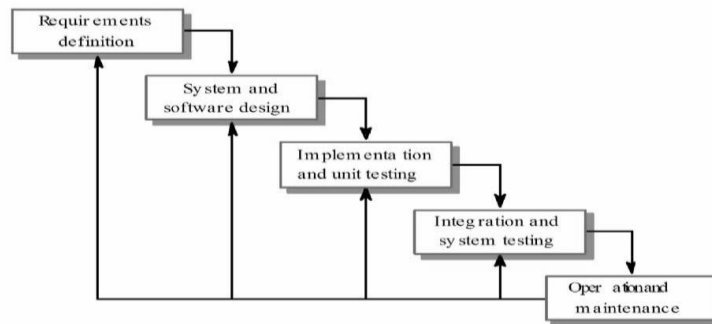
|A| = panjang vektor A

$|B|$  = panjang vektor B

$|A||B|$  = cross product antara  $|A|$  dan  $|B|$

## 2.4 Perancangan Website

Metode pengembangan dalam pembuatan sistem sistem rekomendasi dalam menentukan dosen pembimbing tugas akhir di prodi Informatika UMS menggunakan *Software Development Life Cycle* (SDLC) dengan model waterfall. Metode waterfall memiliki tahapan-tahapan seperti *Requirements analysis and definition*, *System and software design*, *Implementation and unit testing*, *Integration and system testing* dan *Operation and maintenance* (Sommerville, 2011). Tahapan model waterfall ditunjukkan pada Gambar 1.



Gambar 1. Model waterfall

Langkah-langkah di atas dijabarkan lebih lanjut sebagai berikut.

### 2.4.1 Requirements Analysis and Definition

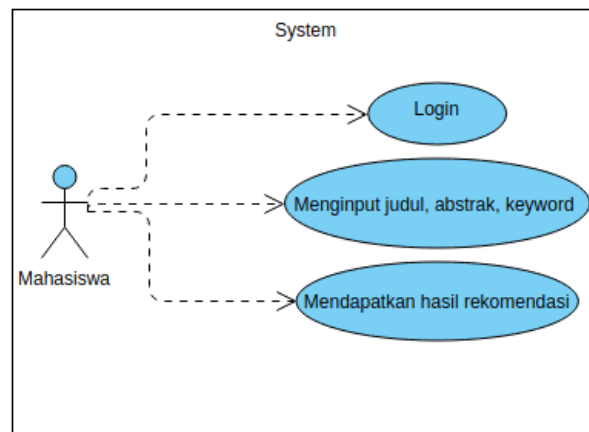
Tahap ini merupakan tahap awal sebelum mengembangkan sebuah sistem, pada tahap ini digunakan untuk mengetahui informasi seperti model aplikasi, spesifikasi yang dibutuhkan dalam mengembangkan aplikasi serta menganalisa kebutuhan fungsional maupun non fungsional. Untuk *tools* yang digunakan dalam pengembangan sistem ini menggunakan *framework* Django yang menggunakan bahasa pemrograman python sebagai *backend* dan menggunakan *framework* ReactJs yang menggunakan bahasa pemrograman javascript sebagai *frontend* dari websitenya.

### 2.4.2 System and Software Design

Tahapan Perancangan sistem juga membantu dalam mengalokasi kebutuhan-kebutuhan sistem baik perangkat keras maupun perangkat lunak dengan membentuk arsitektur sistem secara keseluruhan. Perancangan sistem meliputi *Use Case Diagram*, dan *Activity Diagram*.

a. *Use Case Diagram*

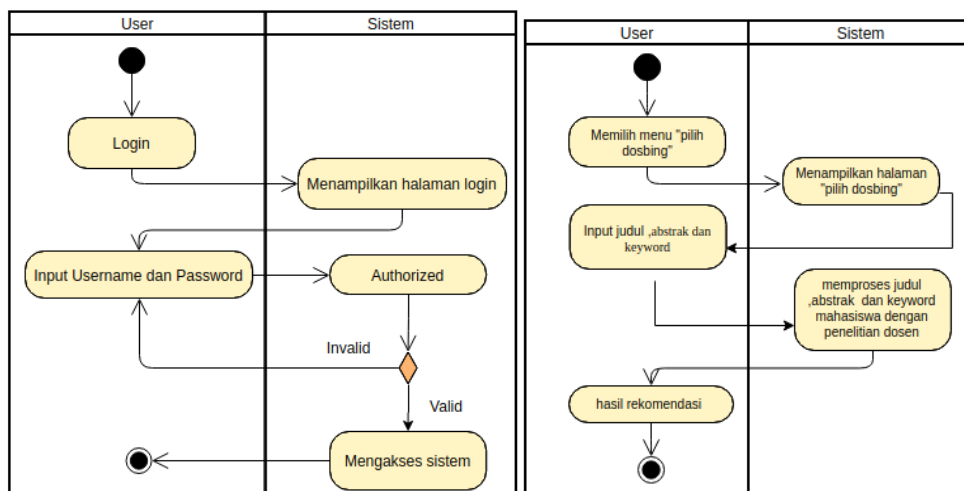
*Use Case Diagram* merupakan gambaran bagaimana *user* dapat berinteraksi dengan sistem. *Use Case Diagram* juga bisa digunakan untuk mengetahui fungsi apa saja yang bisa digunakan dalam sistem. Pada sistem ini terdapat satu aktor yaitu mahasiswa. Mahasiswa bisa melakukan login, menginputkan judul, abstrak dan keyword dari penelitian yang akan dilakukannya serta mendapatkan hasil dari rekomendasi. Seperti ditunjukkan pada Gambar 2.



Gambar 2. *use case diagram*.

b. *Activity Diagram*

Untuk mendokumentasikan kegiatan pengguna, dibuatlah sebuah *activity diagram*. Ini adalah gambaran alur kerja dari proses bisnis dan merupakan urutan aktivitas dalam suatu proses. Sistem ini memiliki beberapa *activity system* seperti *activity login* dan *activity input judul, abstrak, keyword*. Ini ditunjukkan pada Gambar 3.



Gambar 3. *Activity Diagram Login dan Activity Diagram input judul, abstrak dan keyword*.

#### 2.4.3 Implementation and Unit Testing

Pada tahap ini, perancangan perangkat lunak direalisasikan sebagai serangkaian program atau unit program dengan menggunakan beberapa beberapa *library* dan kerangka web akan membantu dalam proses pengembangan menjadi lebih cepat dan lebih konsisten dalam penulisan kode, serta mengurangi redundansi dalam kode. Pada sistem rekomendasi pemilihan dosen pembimbing di prodi informatika UMS memerlukan beberapa alat untuk mengembangkan sistem rekomendasi yang berbasis web, alat yang digunakan dalam penelitian ini adalah python sebagai bahasa pemrograman untuk mengolah data, kemudian Django sebagai *backend* yang berfungsi untuk menyediakan data dan menggunakan ReactJs sebagai *frontend* dari sistem websitenya.

#### 2.4.4 Integration and System Testing

Tahap unit-unit individu program digabung dan diuji sebagai sebuah satu kesatuan sistem yang lengkap untuk memastikan apakah sesuai dengan kebutuhan perangkat lunak atau tidak. Di Tahap ini dilakukan penggabungan antara aplikasi *backend* dengan *frontend* menjadi satu kesatuan sistem website. Kemudian tahap pengujian dilakukan dengan tujuan untuk menemukan kesalahan yang terdapat pada sistem. Dengan melakukan analisa terhadap perangkat lunak kita dapat mengetahui kondisi yang terdapat pada sistem serta mengetahui apa saja yang diperlukan oleh sistem (Ehmer Khan, 2011).

#### 2.4.5 Operation and Maintenance

Tahapan ini merupakan tahapan paling panjang. Sistem dipasang dan digunakan secara nyata. *Maintenance* melibatkan pembetulan kesalahan yang tidak ditemukan pada tahapan-tahapan sebelumnya, meningkatkan implementasi dari unit sistem, dan meningkatkan layanan sistem sebagai kebutuhan baru.

### 3. HASIL DAN PEMBAHASAN

Bagian ini akan mendetailkan tahapan-tahapan pengembangan yang dilakukan beserta hasilnya. Pertama adalah preprocessing, di mana data dibersihkan dan dibentuk menjadi token-token. Sesudah itu, data bersih tersebut diumpankan ke algoritma perekomendasi. Tahapan berikutnya adalah implementasi sistem perekomendasi ini ke sebuah kerangka web.

### 3.1 Tahapan Preprocessing

Sebanyak 603 dataset hasil web scraping dari halaman google scholar yang memiliki 4 atribut. Kemudian dataset tersebut akan direduksi menjadi 2 atribut seperti di Gamba 4, dengan tujuan untuk mempermudah proses preprocessing selanjutnya. Sebelum dijadikan vektor dan diukur kedekatannya, datanya terlebih dahulu dibersihkan dari noise. Selanjutnya data akan melalui proses tokenization. Data yang berupa kalimat panjang akan dipecah menjadi kata-kata atau menjadi sebuah token. Kemudian setelah data menjadi token, data akan masuk ke proses stopwords. Kata-kata yang sering muncul pada dokumen, dan kata-kata tersebut terdaftar dalam stoplist, akan dihilangkan. Misal kata ‘di’, ‘dan’, ‘ke’.

	Nama	Judul	Abstrak	Keyword
0	Heru Supriyono, S.T., M.Sc., PhD	Rancang bangun aplikasi pembelajaran hadis unt...	pada saat ini mobile phone sudah sangat berkem...	Android, pembelajaran ilmu hadis, perangkat mo...
1	Heru Supriyono, S.T., M.Sc., PhD	Pemilihan rumah tinggal menggunakan metode wei...	Pemilihan rumah tinggal termasuk salah satu co...	Pemilihan rumah tinggal; weighted product; sis...
2	Heru Supriyono, S.T., M.Sc., PhD	Rancang bangun media pembelajaran dan game edu...	bahasa Jawa adalah salah satu bahasa daerah ya...	Educative game, interactive learning media, ga...

	Heru Supriyono, S.T., M.Sc., PhD	Dr. Endah Sudarmilah, S.T., M.Eng	Azizah Fatmawati, S.T., M.Cs	Nurgiyatna, S.T., M.Sc., Ph.D	Aris Rakhmadi, S.T., M.Eng	Dedi Gunawan, S.T., M.Sc., Ph.D	Prof. Dr. Budi Murtiyasa, M.Kom	Husni Thamrin, S.T., M.T., Ph.D	Dr., Ir. Bana Handaga, M.T	Fatah Yasin Al Irsyadi, S.T., M.T	Afriyanti Puspita Putri, S.Kom., M.Sc	Devi Anggoro, S.Kom. M.
0	rancang bangun aplikasi pembelajaran hadis unt...	rancang bangun aplikasi pembelajaran hadis unt...	sistem informasi pelanggan pada bengkel marno ...	sistem karpet cerdas berdasarkan fotonik tomog...	perancangan dan pembuatan sistem pintu otomati...	modul pembelajaran interaktif elektronika dasa...	tantangan pembelajaran matematika era global s...	sebuah aturan berbasis swot analysis aplikasi ...	kssk sebuah algoritma untuk mengklasifikasikan...	game edukasi pengenalan anggota tubuh dan peng...	rancang bangun media pembelajaran bahasa arab ...	aplika siste penduku keputusan deng me
1	pemilihan rumah tinggal menggunakan metode wei...	platform permainan untuk kemampuan upgrade men...	prototyping ar edugame untuk anak-anak belajar ...	sistem informasi unit kegiatan mahasiswa march...	perancangan libraryumscms menggunakan codeigni...	evaluasi performa pemecahan database dengan me...	pemanfaatan teknologi informatika dan komunikasi...	sebuah analisis swot alat untuk perusahaan kec...	fuzzy soft set berdasarkan klasifikasi untuk d...	pemanfaatan augmented reality untuk game eduka...	pesma apps sebagai aplikasi terpadu berbasis a...	siste inform manajem administrasi dan ke
2	rancang bangun media pembelajaran dan game edu...	rancang bangun media pembelajaran bahasa dan h...	review benefit serious games game komputer men...	smart carpet imaging objects footprint photoni...	pengendalian kecepatan motor induksi melalui i...	pemanfaatan pemrograman visual sebagai alterna...	isu kunci dan tren penelitian pendidikan ma...	efektivitas algoritma semantik dengan keterkai...	similarity approach fuzzy soft set based numer...	game edukasi pembelajaran bahasa inggris untuk...	game edukasi pembelajaran sejarah berdirinya l...	analisi kepatuh karyaw terhad kebijakar

Gambar 4. Dataset sebelum dan sesudah di *preprocessing*

Setelah data mengalami beberapa proses dan sampai akhirnya data menjadi sebuah token, tahap selanjutnya data akan dijadikan sebuah vektor menggunakan metode TF-IDF dengan n-gram range 1-2 kata. Dalam ilmu linguistik dan probabilitas komputasi, n-gram merupakan urutan yang berdekatan dari n item dari sebuah teks. N-gram akan memberikan probabilitas kemungkinan pada kata berikutnya yang dapat membantu dalam memahami makna suatu teks. Inti dari metode ini adalah melakukan

perhitungan nilai TF dan IDF dari setiap kata kunci terhadap masing-masing dokumen. Nilai TF-IDF dapat dihitung dengan menggunakan persamaan:

Terdapat 2 abstrak sebagai berikut:

d1 = “rancang bangun aplikasi pembelajaran hadist”

d2 = “aplikasi pembelajaran anak usia berbasis”

Misalkan kita ingin menghitung bobot pada kata “anak” pada abstrak d2, karena kata “anak” muncul satu kali pada abstrak d2 maka perhitungan bobot kata “anak” menjadi:

Setelah semua kata dihitung bobotnya maka diperoleh hasil seperti yang ditunjukkan pada Tabel 3, sehingga vektor abstrak d1 adalah [0.0, 0.0, 1.0, 1.0, 1.4, 1.4, 0.0, 1.4, 1.0, 0.0, 1.4, 1.4, 1.4, 0.0, 0.0] dan vektor abstrak d2 adalah [1.4, 1.4, 1.0, 1.0, 0.0, 0.0, 1.4, 0.0, 1.0, 1.4, 0.0, 0.0, 0.0, 1.4, 1.4]. Setelah merubah abstrak d1 dan abstrak d2 menjadi vektor tahap selanjutnya adalah mengukur kedekatan antara vektor yang dibahas lebih detail pada sub-bagian 3.2.

Tabel 3. Hasil perhitungan TF-IDF

Term	tf		df	tf-idf	
	d1	d2		$tf_{i,j} \times \ln(\frac{N+1}{df_i+1}) + 1$	
				d1	d2
anak	0	1	1	0.00	1.40
anak usia	0	1	1	0.00	1.40
aplikasi	1	1	2	1.0	1.0
aplikasi pembelajaran	1	1	2	1.0	1.0
bangun	1	0	1	1.40	0.00
bangun aplikasi	1	0	1	1.40	0.00
berbasis	0	1	1	0.00	1.40
hadist	1	0	1	1.40	0.00
pembelajaran	1	1	2	1.0	1.0
pembelajaran anak	0	1	1	0.00	1.40
pembelajaran hadist	1	0	1	1.40	0.00
rancang	1	0	1	1.40	0.00



rancang bangun	1	0	1	1.40	0.00
usia	0	1	1	0.00	1.40
usia berbasis	0	1	1	0.00	1.40

### 3.2 Model Recommendation System

Data yang telah menjadi vektor akan diukur kedekatannya dengan vektor hasil inputan judul, abstrak dan keyword dari mahasiswa. Pada penelitian ini pengukuran kedekatan antar vektor dihitung dengan menggunakan metode *cosine similarity*, merupakan metode yang digunakan untuk mengukur similaritas antar vektor. Pada tahap sebelumnya telah menghasilkan nilai vektor dari setiap penelitian dosen akan mewakili vektor A dan nilai dari setiap inputan mahasiswa akan mewakili vektor B

vektor A = [0.0, 0.0, 1.0, 1.0, 1.4, 1.4, 0.0, 1.4, 1.0, 0.0, 1.4, 1.4, 1.4, 0.0, 0.0]

vektor B = [1.4, 1.4, 1.0, 1.0, 0.0, 0.0, 1.4, 0.0, 1.0, 1.4, 0.0, 0.0, 0.0, 1.4, 1.4]

Setelah semua vektor dihitung kedekatannya maka diperoleh hasil seperti yang ditunjukkan pada Tabel 4,

Tabel 4. Hasil perhitungan *cosine similarity*

No	Nama dosen	Nilai <i>cosine similarity</i>
1	Fatah Yasin Al Irsyadi, S.T., M.T	0.53
2	Aris Rakhmadi, S.T., M.Eng	0.51
3	Fajar Suryawan, Ph.D	0.45
4	Maryam, S.Kom., M.Eng	0.30
5	Nurghiyatna, S.T., M.Sc., Ph.D	0.20

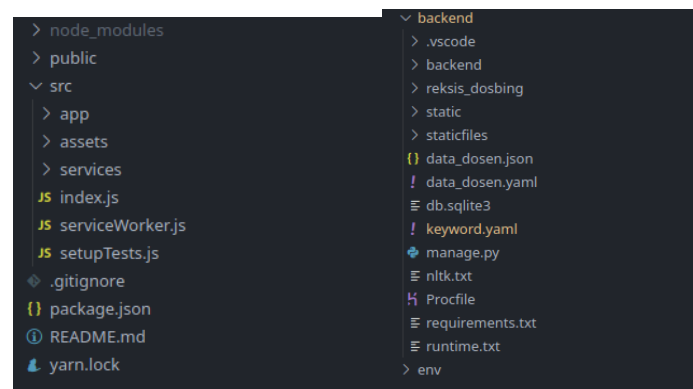
Kemudian hasil pengukuran dari vektor-vektor tersebut akan diurutkan berdasarkan nilai *cosine similarity*-nya. Jika nilai *cosine similarity*-nya mendekati 1, maka vektor tersebut mempunyai kecenderungan similar dengan vektor hasil inputan mahasiswa, dan jika nilai *cosine similarity* mendekati 0, maka vektor tersebut mempunyai kecenderungan tidak similar dengan vektor hasil inputan mahasiswa. Setelah didapatkan hasil pengurutan, tahap selanjutnya adalah membungkus semua

proses perhitungan *cosine similarity* dan proses sorting ke dalam sebuah model class python yang nanti akan dipasang ke dalam sistem website.

### 3.3 Implementasi Sistem

Sistem *website* dirancang sebagai antarmuka untuk mahasiswa, sekaligus digunakan sebagai implementasi sistem rekomendasi untuk memilih dosen pembimbing tugas akhir. *Flow* sistem berjalan satu arah dimulai dari mahasiswa menginputkan judul, abstrak dan keyword kemudian sistem akan melakukan perhitungan untuk menghasilkan dosen rekomendasi yang cocok berdasarkan inputan dari mahasiswa.

Sistem website dibangun menggunakan Django sebagai *backend* dan Reactjs sebagai *frontend*. Di dalam Django terdapat struktur direktori *project* yang seperti yang ditunjukkan pada gambar 5, masing-masing direktori mempunyai fungsi tersendiri, direktori backend sebagai *base project* yang berisi config dan setingan apps Django, kemudian dalam direktori reksis\_backend berfungsi sebagai apps yang akan menangani *request-request* dan pengolahan data dari *frontend*. Demikian juga pada ReactJs juga terdapat struktur direktori yang mempunyai fungsi tersendiri, direktori node\_modules berfungsi sebagai tempat untuk menyimpan *package* yang dibutuhkan oleh ReactJs seperti, bootstrap, multiple select, redux, dll. Kemudian direktori public digunakan untuk menyimpan assets seperti image, icon, dan file html, yang terakhir direktori src, pada direktori ini terdapat file-file javascript yang berguna untuk menghandle komponen-komponen yang dibutuhkan dalam pembuatan *user interface*.



Gambar 5. Struktur direktori ReactJs dan Django

Pada Tabel 5 terdapat 5 *endpoint* yang terdiri dari 2 method POST dan 3 method GET setiap *endpoint* mempunyai fungsi masing-masing, *endpoint* rest-auth/google berfungsi sebagai jalur untuk menggunakan OAuth2 google sebagai autentikasi untuk masuk ke dalam website. Kemudian *endpoint* api/keyword berfungsi untuk

menyediakan keyword-keyword yang akan dipakai oleh mahasiswa, terdapat 2466 keyword yang berasal dari penelitian dosen pembimbing yang terdapat di halaman google scholar, selanjutnya *endpoint* api/reksis berfungsi sebagai jalur untuk memproses inputan data dari mahasiswa dengan model rekomendasi sistem.

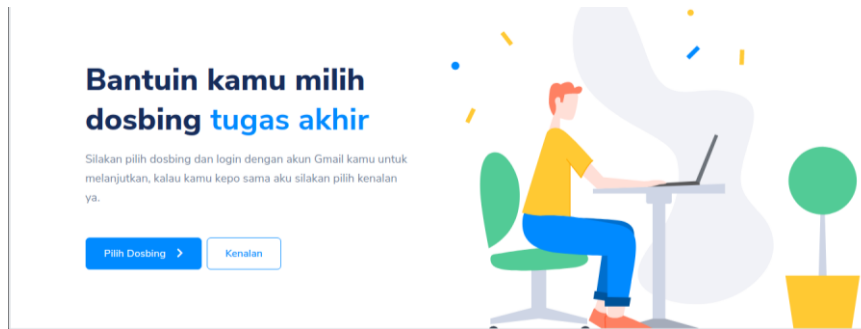
Tabel 5. Daftar *endpoint* di backend

<b>Method</b>	<b>Endpoint</b>	<b>Keterangan</b>
GET	api/keyword	Menyediakan data keyword
GET	api/dosen	Menyediakan data dosen pembimbing
POST	api/reksis	Melakukan pengolahan data dan menyediakan data hasil rekomendasi
POST	api/auth/google	Login dengan OAuth2 Google
GET	api/auth/logout	Logout dari sistem

Tabel 6. Daftar *routes* di frontend

<b>Method</b>	<b>Route</b>	<b>Keterangan</b>
GET	/	Menampilkan halaman utama
GET	/reksis	Pengisian data oleh user dan menampilkan hasil rekomendasi
GET	/dosbing	Menampilkan daftar dosen pembimbing
GET	/about	Menampilkan halaman about

Gambar 6 merupakan halaman utama ketika website diakses oleh mahasiswa. Halaman ini diakses menggunakan route /, di halaman ini akan ada dua tombol, tombol “pilih dosbing” dan tombol “kenalan” jika mahasiswa akan memilih tombol “kenalan” mahasiswa akan di-redirect ke halaman yang berisi pengenalan terhadap sistem, dan jika mahasiswa memilih tombol “pilih dosbing” kemudian akan di redirect ke sistem OAuth2 google, untuk melakukan autentifikasi.



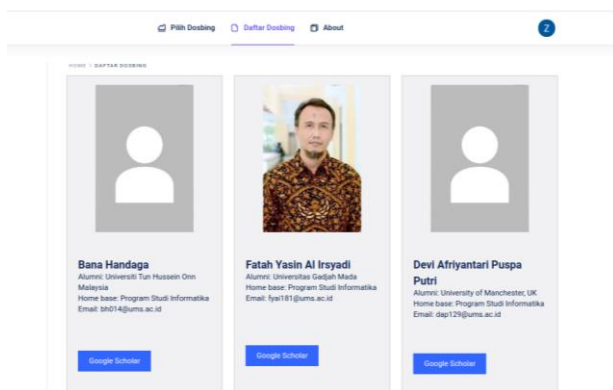
Gambar 6. Halaman utama website

*Route* /reksis berfungsi untuk menangani ketika mahasiswa berhasil login, sekaligus menangani mahasiswa dalam melakukan pengisian form untuk dan menentukan rekomendasi dosen pembimbing tugas akhir, serta menampilkan hasil dari sistem rekomendasinya.

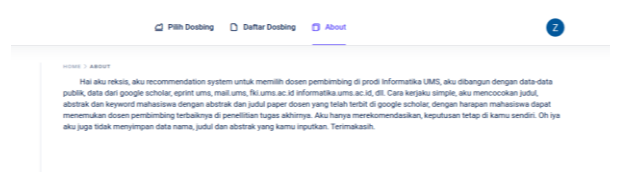
Gambar 7. Halaman input judul, abstrak dan keyword

Gambar 8. Halaman hasil dari rekomendasi dosen pembimbing

*Route* /dosbing berfungsi untuk menampilkan daftar dosen pembimbing tugas akhir yang tersedia, kemudian *route* /about berfungsi untuk menampilkan tulisan tentang data-data yang dipakai oleh sistem rekomendasi, termasuk sumber data nya dan sekilas cara kerja sistem rekomendasinya.



Gambar 9. Halaman daftar dosen pembimbing tugas akhir



Gambar 10. Halaman about website

Tahap pengujian sistem merupakan tahap terakhir yang berfokus pada hasil akhir dan fitur-fitur yang terdapat dalam sistem, Pengujian sistem dengan *black box* merupakan pengujian dengan sistem langsung dihadapkan dengan pengguna untuk berinteraksi dan sistem mampu merespon dengan baik dan sesuai yang direncanakan.

Tabel 6. *Black Box Testing*

Fungsi	Input	Output	Status
Halaman utama	Mengakses website	Menampilkan halaman utama	Valid
Halaman login	Menginputkan Gmail dan Password	Menampilkan halaman untuk pengisian form sistem rekomendasi	Valid
Halaman rekomendasi	Menginputkan data seperti judul, abstrak dan keyword	menampilkan hasil rekomendasi dosen pembimbing	Valid
Halaman daftar dosen pembimbing	Mengakses halaman daftar dosen pembimbing	Menampilkan daftar dosen pembimbing	Valid

## 4. PENUTUP

### 4.1 Kesimpulan

Dari hasil penelitian dan perancangan yang telah dilakukan dapat diketahui bahwa data hasil dari *web scraping* masih merupakan data mentah dan kotor, masih harus melewati berbagai proses terlebih dahulu sebelum data dapat benar-benar dimanfaatkan. Kemudian secara fungsionalitas sistem rekomendasi secara umum berfungsi sesuai yang

telah direncanakan. Di prodi Informatika UMS terdapat dosen yang mempunyai kecenderungan minat dan keahlian di bidang jaringan, sistem informasi, *game development*, *data scientist*, dan sistem rekomendasi ini juga akan merekomendasikan dosen tersebut jika diberikan inputan topik sesuai dengan minat dan keahlian dari dosen. Dengan demikian dapat disimpulkan bahwa penelitian ini sesuai dengan tujuan.

Sistem rekomendasi dengan metode *Content-based filtering* akan sangat bergantung pada konten yang ada pada setiap item, semakin banyak konten didalam item maka semakin baik pula hasil dari rekomendasinya, tetapi banyak konten pada metode *Content base filtering* juga akan berpengaruh pada waktu eksekusi, semakin besar konten maka semakin besar pula waktu yang dibutuhkan oleh sistem untuk melakukan perhitungan.

*Content-based filtering* juga mempunyai kelemahan, karena *Content-based filtering* sangat bergantung dengan konten pada item, maka *Content-based filtering* tidak dapat menghasilkan rekomendasi yang sesuai jika konten yang dianalisis untuk suatu item tidak mengandung informasi yang sesuai untuk kategorisasi, atau pada item tidak mempunyai cukup konten untuk melakukan kategorisasi.

#### **4.1 Saran**

Saran untuk penelitian selanjutnya di bidang sistem rekomendasi yaitu supaya lebih bisa memanfaatkan data-data yang bersifat publik seperti data repository perpustakaan kampus untuk membangun sistem rekomendasi yang baik serta untuk peneliti selanjutnya diharapkan juga menggunakan algoritma yang lain dalam menentukan hasil rekomendasi suatu item untuk memaksimalkan hasil dari rekomendasi.

#### **DAFTAR PUSTAKA**

- Abdullah, A., & Pangestika, M. W. (2018). Perancangan Sistem Pendukung Keputusan dalam Pemilihan Dosen Pembimbing Skripsi Berdasarkan Minat Mahasiswa dengan Metode AHP (Analytical Hierarchy Process) di Universitas Muhammadiyah Pontianak. *JEPIN (Jurnal Edukasi dan Penelitian Informatika)*, 4(2), 184–191. <https://doi.org/10.26418/jp.v4i2.27651>
- Amalina, N., & Sutikno, S. (2017). *Sistem Rekomendasi Dosen Pembimbing Tugas Akhir Berbasis Text Mining Menggunakan Vector Space Model* [Other, Universitas Diponegoro]. <http://eprints.undip.ac.id/60645/>
- Debnath, S., Ganguly, N., & Mitra, P. (2008). Feature weighting in content based recommendation system using social network analysis. *Proceeding of the 17th*

- International Conference on World Wide Web - WWW '08*, 1041.  
<https://doi.org/10.1145/1367497.1367646>
- Ehmer Khan, Mohd. (2011). Different Approaches To Black box Testing Technique For Finding Errors. *International Journal of Software Engineering & Applications*, 2(4), 31–40. <https://doi.org/10.5121/ijsea.2011.2404>
- Gunawan, D. (2016). Evaluasi Performa Pemecahan Database dengan Metode Klasifikasi Pada Data Preprocessing Data mining. *Khazanah Informatika: Jurnal Ilmu Komputer dan Informatika*, 2(1), 10–13. <https://doi.org/10.23917/khif.v2i1.1749>
- Jianqiang, Z., Xiaolin, G., & Xuejun, Z. (2018). Deep Convolution Neural Networks for Twitter Sentiment Analysis. *IEEE Access*, 6, 23253–23260. <https://doi.org/10.1109/ACCESS.2017.2776930>
- Mubarak, F. (2019). Perbandingan Cosine Similarity dan Euclidean Distance pada Sistem Rekomendasi Film Menggunakan Metode Item Based Multi Criteria Collaborative Filtering. *Universitas Sebelas Maret*.
- Nagarnaik, P., & Thomas, A. (2015). Survey on recommendation system methods. *2015 2nd International Conference on Electronics and Communication Systems (ICECS)*, 1603–1608. <https://doi.org/10.1109/ECS.2015.7124857>
- Nilashi, M., Bagherifard, K., Ibrahim, O., Alizadeh, H., Nojeem, L. A., & Roozegar, N. (2013). Collaborative Filtering Recommender Systems. *Research Journal of Applied Sciences, Engineering and Technology*, 5(16), 4168–4182. <https://doi.org/10.19026/rjaset.5.4644>
- Pyle, D. (1999). *Data Preparation for Data Mining*. 466.
- Rahutomo, F., Kitasuka, T., & Aritsugi, M. (2012). *Semantic Cosine Similarity*.
- Robertson, S. (2004). Understanding Inverse Document Frequency: On Theoretical Arguments for IDF. *Journal of Documentation - J DOC*, 60, 503–520. <https://doi.org/10.1108/00220410410560582>
- Roul, R. K., Sahoo, J. K., & Arora, K. (2017). Modified TF-IDF Term Weighting Strategies for Text Categorization. *2017 14th IEEE India Council International Conference (INDICON)*, 1–6. <https://doi.org/10.1109/INDICON.2017.8487593>
- Samuel, R., Natan, R., & Syafiqoh, U. (2018). *Penerapan Cosine Similarity dan K-Nearest Neighbor (K-NN) pada Klasifikasi dan Pencarian Buku*. 1(1), 6.
- Sharma, L., & Gera, A. (2013). A Survey of Recommendation System: Research Challenges. *International Journal of Engineering Trends and Technology*, 5.
- Sommerville, I. (2011). *Software engineering* (9th ed). Pearson.
- Son, J., & Kim, S. B. (2017). Content-based filtering for recommendation systems using

multiattribute networks. *Expert Systems with Applications*, 89, 404–412.  
<https://doi.org/10.1016/j.eswa.2017.08.008>

Zhang Yun-tao, Gong Ling, W. Y. (2005). *An improved TF-IDF approach for text classification*. <https://link.springer.com/article/10.1007%2F02842477>